



DELETE con Statement

Transcripción

[00:00] Hola, nuestra aplicación va quedando más completa y con eso vamos aprendiendo más cómo realizar cada tipo de operación de base de datos con Java. Acabamos de aprender cómo realizar un insert y ahora vamos a aprender cómo eliminar un producto haciendo un delete.

[00:15] Aquí en nuestra aplicación, vamos a agregar un nuevo producto. Vamos a agregar un celular, y en la descripción va a ser un celular Samsung de 50 unidades. Haga un clic en guardar. Tengo el teléfono ya agregado acá en nuestra tabla, en el listado, pero ahora tenemos acá, ah, ya tenía un celular Samsung, me equivoqué, perdón debía haber agregado un celular Samsung acá.

[00:47] Entonces vamos a hacer lo siguiente, ya que tenemos acá el eliminar, vamos a hacer un clic acá y hacemos un clic en eliminar. Un montón de error en la consola y nada funciona. O sea, es porque no tenemos nada desarrollado todavía. Entonces en esta clase vamos a enfocarnos en cómo vamos a eliminar este nuevo registro que agregamos por equivoco.

[01:11] Bueno, vamos acá al Eclipse para ver qué pasó primero con este log, porque eso es raro, ¿no? No esperaba que hubiera un error ya de cara así, un error muy raro. Vamos a ver acá, bueno, tenemos acá un ClassCastException, no podemos tomar un string no puede ser casteado para el tipo integer. ¿En dónde? Bueno, acá en el ControlDeStockFrame hago un clic acá en esta línea.

[01:35] Me lleva justamente a donde hubo el error. ¿Y qué nos dijo el log? ¿El log dijo que no pudimos hacer un cast, el tipo string para el tipo integer. ¿Será que es porque estamos haciendo un cast explícito acá? Estoy haciendo el paréntesis, integer y tomó el valor del getValueAt. Bueno, puede ser eso tal vez, entonces vamos a hacer lo siguiente, vamos a cambiar eso acá.

[02:02] En lugar de hacer un cast para integer, vamos a hacer Integer.valueOf, y ahora sí ponemos este valor adentro como un parámetro. Pero ahora el Eclipse se está quejando de algo. Vamos a ver qué dice. Bueno dice que es un valueOf the string, y nosotros estamos enviando un object, o sea, tenemos que hacer un cambio más acá.

[02:28] Entonces vamos a hacer acá en el getValueAt, este elemento acá de la línea elegida de la columna 0, vieron que en Java una lista empieza con 0. Hacemos un .toString(); y ahora sí, todo bien. El Eclipse no se queja más, o sea, parece que ya funciona. Vamos a ver acá, voy a limpiar la consola, volvemos a nuestra pantalla acá.

[02:57] Todo bien, que todavía no tenemos la funcionalidad de eliminar, pero primero vamos a arreglar este error en el log y ahí seguimos con la lógica de SQL. Acá la línea 4 elegida botón eliminar. Sigue el error. ¿Por qué? Porque no hice el build de la aplicación y no reinicié la aplicación. Entonces voy a parar acá la aplicación, ahora sí vengo a ControlDeStockMain, hago un run as, Java application y ahora sí vamos a probar.

[03:36] Bueno, ahora vamos a ver si finalmente sale el log, ahora sí. Item eliminado con éxito, desapareció del listado pero con un okay, vuelve otra vez, porque aún no la eliminamos de hecho, en la base de datos, ahora sí vamos para la lógica de eliminar. Bueno, para saber en dónde tenemos que agregar la lógica, vamos a hacer lo mismo que hicimos con las otras funcionalidades.

[04:00] Vamos a ver el camino del botón ahora de eliminar, como hicimos con el guardar. Vamos a ver acá AccionesDelFormulario y vamos a buscar el botón

eliminar y ver que él hace un eliminar, el método eliminar, limpia la tabla y la carga una vez más.

[04:18] Entonces vamos acá con eliminar, tenemos acá una lógica de validación, tenemos acá la parte que arreglamos, que fue el casteo, de la conversión de string para un integer, y estamos acá con eliminar que envía el id del producto. Entonces va a ser acá en este método eliminar que vamos a entrar.

[04:41] Acá entramos y justamente está el comentario de to do. Es acá que vamos a agregar esta lógica para hacer el delete del producto. Bueno, entonces para empezar, voy a borrar el comentario y como siempre hacemos, ya copio acá que queda fácil. Voy a tomar una conexión y antes de seguir ya vamos a hacer una throws SQLException y en el ControlDeStockFrame vamos a agregar eso adentro de un bloque de try catch.

[05:12] Y acá encapsulamos esta excepción en un throw new RuntimeException enviando como parámetro la excepción. Listo. Ya Eclipse no va a quejarse más de eso. Ahora sí podemos seguir con nuestra lógica. Bueno, ¿qué era el próximo paso? Hacer un statement. Entonces con.createStatement(); acá lo asignamos a una variable del tipo también statement. Ahí está.

[05:47] Y en este statement vamos a hacer el execute. Entonces acá vamos a escribir statement.execute y como parámetro, ponemos el string "DELETE FROM PRODUCTO WHERE ID = "+ id); Ahí está, finalizamos. Me parece que sí, pero vamos a garantizar que todo corrió bien. Y para saberlo, para saber si algo fue realmente eliminado, hay un método acá que es el statement.executeUpdate.

[06:28] Y esto acá nos va a devolver un Int. Entonces acá voy asignarlo a una variable que es un UpdateCount, y este número del tipo Int nos devuelve cuántas filas fueron modificadas luego que ejecutamos el comando de SQL en

el statement, o sea aquí en este caso, como estamos haciendo un delete de un id específico, ¿entonces cuántos registros vamos a borrar?

[06:56] Acá en este caso va a ser uno. Vamos a ver acá, entonces, qué nos dice acá en la aplicación. Pero en lugar de estar haciendo un `system.out.println` vamos a hacer algo distinto ahora, vamos a hacer que la aplicación en el formulario nos diga cuántos registros fueron eliminados. Entonces, para hacerlo, nosotros vamos a hacer lo siguiente.

[07:17] Vamos a hacer un return, en lugar de asignar a una variable, vamos a hacer un return de `UpdateCount`, y acá vamos a estar devolviendo también un `int` en el método. Okay, bueno, acá estamos retornando y ahora en el `ControlDeStockFrame` en el `productoController.eliminar` nosotros vamos a asignar este valor a una variable también, una variable de `cantidadEliminada`.

[07:46] Pero eso tiene que estar afuera porque nosotros vamos a decir cuántos elementos fueron eliminados. Entonces voy acá a sacar esta variable, la voy a dejar acá declarada y acá pongo solamente la asignación. Y en el mensaje de `JOptionPane`, acá, nosotros vamos a decir que `cantidadEliminada` espacio "item eliminado con éxito!"; okay, vamos a guardar todo.

[08:26] Voy a bajar la aplicación, la voy a ejecutar una vez más y vamos a hacer las pruebas. Listo, acá está la aplicación cargada, vamos a borrar entonces esta fila. Elegí acá la fila que quiero eliminar, hago un clic en eliminar y acá dice un item eliminado con éxito, hago un okay y no cargó más. Voy a reiniciar la aplicación.

[08:52] Mira, la aplicación se reinició y seguimos acá con los tres productos, o sea, el delete funciona. Pero ahora una duda. ¿Qué pasaría? ¿Cuál sería el resultado de `getUpdateCount` si ninguna fila hubiera sido modificada? Vamos a hacer una prueba. [09:09] Acá en el Eclipse vamos a el paquete acá que tenemos de pruebas, vamos a crear una nueva clase. La clase va a llamarse acá primero `class`, el hijo, y la clase va a llamarse `PruebaDelete`. Hago un finish acá.

Y hagamos lo siguiente, voy a crear un main acá primero. Y vamos acá en el `productoController` a hacer una copia de esta lógica de acá.

[09:37] Y recién escribimos de `productoController` para hacer el delete. Y acá vamos a hacer lo siguiente en lugar del id que nosotros no tenemos, vamos a poner `ID = 99` y acá si vamos a hacer un `system.out.println` para saber la cantidad que fue eliminada. Acá hacemos ahora un `throws exception`, ahí está, y vamos a ejecutarlo.

[10:08] Bueno, sabemos que el `ID = 99` no existe en la base de datos, entonces el resultado esperado debería ser 0. Vamos a ver qué pasa. Run as Java application, y acá en la consola, a ver, bueno, devuelve 0, o sea, eso indica que ningún cambio hubo ahí en la tabla. Listo, ya tenemos tres funcionalidades completas en la aplicación, donde ya pudimos aprovechar para aprender a realizar las operaciones de SQL con Java y crear un patrón de diseño.

[10:41] Esas operaciones desde SQL fueron el select, el insert y el delete. La funcionalidad de modificar un registro con el comando de update va a quedar como un desafío para ustedes. ¿Pensaron que sería solamente estar viendo acá a los vídeos? Ahora la acción queda para ustedes.

[11:01] O sea, nosotros vamos a agregar un nuevo producto acá en el depósito. Tengo acá el producto y quiero escribir, voy a agregar una cuchara. Y una cuchara de metal. Cuchara de metal, 100 unidades. Guardé. Ahí registré la cuchara pero escribí acá cucaracha, me fue la palabra. Bueno, acá, yo podría hacer la corrección cuchara, y cuando hago un clic en modificar, tendría que modificarla sin ningún problema.

[11:48] Entonces este va a ser el desafío que queda para ustedes. Ya vemos acá que cuando hice clic en modificar, ya tenemos un errorcito acá en el log que va a ser el mismo de `ClassCastException`, entonces que donde desafío más completo para ustedes, para tratar un error y para crear una funcionalidad.

[12:06] Entonces la parte divertida queda para ustedes ahora. Nos vemos en la próxima clase.